

CLAIMS

WE CLAIM:

1. A display driver to display a file stream, comprising:
 - a display driver module having a bitmapped frame buffer, the display driver module controlling the display; and
 - a decoder to transform the file stream and store the transformed file stream in the bitmapped frame buffer of the display driver module, the display driver adapted to process data in the bitmapped frame buffer to generate the display.
2. The display driver of claim 1 wherein the display driver module and decoder are disposed on a same substrate.
3. The display driver of claim 1 wherein the display driver is adapted to perform the steps comprising:
 - determining if a user has authorization if digital rights management has been applied to the file stream; and
 - if the user has authorization, performing the steps of transforming the file stream and storing the transformed file stream in the bitmapped frame buffer.
4. The display driver of claim 3 wherein the display driver is further adapted to perform the step of decrypting the file stream if the file stream is encrypted.

5. The display driver of claim 1 wherein the file stream contains metadata, the display driver further comprising a processor to process metadata from the file stream.

6. The display driver of claim 1 wherein decoder is adapted to transform the file stream from a MPEG-2 format into the bitmapped frame buffer of the display driver module.

7. The display driver of claim 1 wherein decoder is adapted to transform the file stream from a Windows Media File (WMF) format into the bitmapped frame buffer of the display driver module.

8. The display driver of claim 1 wherein decoder is adapted to transform the file stream from a next generation MPEG compression scheme format into the bitmapped frame buffer of the display driver module.

9. The display driver of claim 1 wherein the display driver is adapted to process data in the bitmapped frame buffer to generate a Digital Light Processing display.

10. The display driver of claim 1 wherein the display driver is adapted to process data in the bitmapped frame buffer to generate a Liquid Crystal Device (LCD) display.

11. The display driver of claim 1 wherein the display driver is adapted to process data in the bitmapped frame buffer to generate a command signal to drive a Micro Electrical Mechanical (MEM) controlled rendering device.

12. A method to drive a display driver of an encoded file stream comprising the steps of:

receiving the encoded file stream;

transforming the encoded file stream into a format of the display driver, thereby generating a transformed file stream; and

storing the transformed file stream in the bitmapped frame buffer of the display driver.

13. The method of claim 12 further comprising the step of decoding the encoded file stream.

14. The method of claim 12 further comprising the step of processing data in the bitmapped frame buffer to generate a display.

15. The method of claim 14 wherein the step of processing data in the bitmapped frame buffer to generate a display comprises the step of processing data in the bitmapped frame buffer to generate a Digital Light Processing display.

16. The method of claim 14 wherein the step of processing data in the bitmapped frame buffer to generate a display comprises the step of processing data in the bitmapped frame buffer to generate a Liquid Crystal Device (LCD) display.

17. The method of claim 12 wherein the step of processing data in the bitmapped frame buffer to generate a display comprises the step of processing data in the bitmapped frame buffer to generate a command signal to drive a Micro Electrical Mechanical (MEM) controlled device.

18. The method of claim 12 wherein steps are performed on a same substrate.

19. The method of claim 12 further comprising the steps of:
determining if a user has authorization if digital rights management has been applied to the file stream;
if the user has authorization, performing the steps of transforming the file stream into a format of the display driver module and storing the transformed file stream in the bitmapped frame buffer; and
dropping the file stream without performing the steps of transforming the file stream into a format of the display driver module and storing the transformed file stream in the bitmapped frame buffer if the user does not have authorization

20. The method of claim 19 further comprising the step of decrypting the file stream if the file stream is encrypted.

21. The method of claim 12 wherein the file stream contains metadata, the method further comprising the step of processing the metadata.
22. The method of claim 12 wherein the step of transforming the encoded file stream into a format of the display driver module comprises the step of transforming a MPEG-2 encoded file stream into the bitmapped frame buffer of the display driver module.
23. The method of claim 12 wherein the step of transforming the encoded file stream into a format of the display driver module comprises the step of transforming a Windows Media File (WMF) encoded file stream into the bitmapped frame buffer of the display driver module.
24. The method of claim 12 wherein the step of transforming the encoded file stream into a format of the display driver module comprises the step of transforming a next generation MPEG compression scheme encoded file stream into the bitmapped frame buffer of the display driver module.
25. A method to apply digital rights management of data from the point of capture to the point of rendering comprising the steps of:
 - capturing the data;
 - storing the data directly into a frame buffer of an encoder;

transforming the data in the frame buffer into an encoded media file;
applying digital rights management to the encoded media file;
transmitting the encoded media file to a rendering device;
unwrapping the digital rights management applied to the encoded media file;
decoding the encoded media file into a driver frame buffer; and
generating commands to control display components using data in the driver
frame buffer.

26. The method of claim 25 further comprising the step of sending the
commands to the rendering components.

27. The method of claim 25 wherein the steps of capturing data, storing the
data directly into a frame buffer of an encoder, transforming the data in the frame buffer
into an encoded media file, and applying digital rights management to the encoded media
file includes performing the steps of capturing data, storing the data directly into a frame
buffer of an encoder, transforming the data in the frame buffer into an encoded media
file, and applying digital rights management to the encoded media file on a same
substrate.

28. The method of claim 27 wherein the steps of unwrapping the digital rights
management applied to the encoded media file, decoding the encoded media file into a
display driver frame buffer, generating commands to control display components based
on data in the driver frame buffer, and sending the commands to the display components

includes performing the steps of unwrapping the digital rights management applied to the encoded media file, decoding the encoded media file into a driver frame buffer, generating commands to control display components based on data in the driver frame buffer, and sending the commands to the display components on a second substrate.

29. The method of claim 25 wherein the steps of unwrapping the digital rights management applied to the encoded media file, decoding the encoded media file into the display driver frame buffer, generating commands to control display components based on data in the driver frame buffer, and sending the commands to the display components includes performing the steps of unwrapping the digital rights management applied to the encoded media file, decoding the encoded media file into the driver frame buffer, generating commands to control display components based on data in the driver frame buffer, and sending the commands to the display components on a same substrate.

30. The method of claim 25 wherein the step of transforming the data in the frame buffer into an encoded media file comprises transforming the data in the frame buffer into a MPEG-2 encoded media file and the step of decoding the encoded media file into the driver frame buffer comprises the step of decoding the MPEG-2 encoded media file into the driver frame buffer.

31. The method of claim 25 wherein the step of transforming the data in the frame buffer into an encoded media file comprises transforming the data in the frame buffer into a Windows Media File (WMF) encoded media file and the step of decoding

the encoded media file into the driver frame buffer comprises the step of decoding the WMF encoded media file into the driver frame buffer.

32. The method of claim 25 wherein the step of transforming the image data in the frame buffer into an encoded media file comprises transforming the image data in the frame buffer into a next generation MPEG compression scheme encoded media file and the step of decoding the encoded media file into the driver frame buffer comprises the step of decoding the next generation MPEG compression scheme encoded media file into the driver frame buffer.

33. The method of claim 25 further comprising the step of applying metadata contained in the encoded media file.

34. The method of claim 25 wherein the step of generating commands to control display components comprises the step of generating commands to control Digital Light Processing (DLP) components.

35. The method of claim 25 wherein the step of generating commands to control display components comprises the step of generating commands to control Liquid Crystal Device (LCD) components.

36. The method of claim 25 wherein the step of generating commands to control display components comprises the step of generating commands to control a Micro Electrical Mechanical (MEM) controlled device.